Express Mailing Label No. EV 328 472 405 US

PATENT APPLICATION
Docket No.: 13768.373

UNITED STATES PATENT APPLICATION

of

Erik Olson
David S. Ebbo
Susan M. Warren
Nikhil Kothari
Scott Guthrie
Peixin Han

and

Dmitry Robsman

for

SYSTEMS AND METHODS FOR MITIGATING CROSS-SITE SCRIPTING

SYSTEMS AND METHODS FOR MITIGATING CROSS-SITE SCRIPTING

BACKGROUND OF THE INVENTION

The Field of the Invention

[0001] The present invention relates to systems and methods for protecting servers and clients from attacks. More particularly, the present invention relates to systems and methods for mitigating cross-site scripting attacks in web applications.

Background and Relevant Art

[0002] The Internet is a network that provides access to significant resources and activities. When a user visits different web sites over the Internet, information is exchanged between the user and the various web sites. Some of the information (account numbers, user names, passwords, personal information, etc.) exchanged between a user and a web site is more confidential than other information. In fact, some of the information is necessary in order to perform certain activities. In addition, the information exchanged between a user and a web site is often stored on the user computer.

[0003] Web-servers as well as user computers have various security concerns and the exchange of information between a user computer and a web site is one or the reasons that security is required to protect the information. One of the more common security concerns is cross-site scripting. Cross-site scripting attacks typically occur in scenarios where a server generates a dynamic web page. By creating a dynamic web page, the server may relinquish control over how the output is interpreted by the user computer. In a cross-site scripting attack, a security issue arises if untrusted dynamic content can be introduced into a dynamic page.

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

[0004] A cross-site scripting attack occurs, for example when an attacker creates an HTTP request that echoes back unencoded or unescaped client script provided by the attacker. The user computer then executes the script in the security context of the origin web server. This may permit the attacker to steal, for example, authentication tokens and private information. A cross-site cross scripting attack may also permit an attacker to run commands with privilege on the user computer being attacked.

[0005] On the Internet, many web-servers are unknowingly vulnerable to cross-site scripting attacks. Even though cross-site scripting attacks can be practically eliminated by rigorously validating and encoding data, many developers do not have the experience or knowledge to do this effectively. In addition, an approach that encodes all output has an impact on performance and may destroy data by encoding previously encoded data. There is a need for systems and methods that mitigating cross-site scripting attacks.

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

BRIEF SUMMARY OF THE INVENTION

[0006] These and other limitations are overcome by the present invention, which relates to systems and methods for mitigating cross-site scripting attacks. Cross-site scripting is a server-side weakness that is realized when user input is rendered as HTML containing active script. When a client computer processes the response from the server, the client/user browser executes the script. Compromised user data is one result of cross-site scripting.

[0007] The present invention mitigates cross-site scripting attacks by evaluating HTTP requests at the server before dynamic rendering of the response is performed. In one embodiment, this is done by examining the HTTP request for script constructs that may indicate a cross-site scripting attack. If a script construct is discovered, then the server computer can abort the HTTP request before the cross-site scripting attack can be completed. The server, for example, can generate an error instead of serving the HTTP request, generate an error event, and the like.

[0008] Typically, the server computer only needs to examine the portions of the HTTP request that may contain data derived from outside input. The portions of the HTTP request at risk are examined for the presence of script constructs. In addition to searching for script constructs or indicators such as angle brackets, the present invention also searches for other script constructs that are only harmful when rendered inside of particular HTML elements. Elements that could be used in a cross-site scripting attack include, but are not limited to, events, expressions, and the like.

[0009] Additional features and advantages of the invention will be set forth in the description which follows, and in part will be obvious from the description, or may be learned by the practice of the invention. The features and advantages of the invention

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

may be realized and obtained by means of the instruments and combinations particularly pointed out in the appended claims. These and other features of the present invention will become more fully apparent from the following description and appended claims, or may be learned by the practice of the invention as set forth hereinafter.

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] In order to describe the manner in which the above-recited and other advantages and features of the invention cam be obtained, a more particular description of the invention briefly described above will be rendered by reference to specific embodiments thereof which are illustrated in the appended drawings. Understanding that these drawings depict only typical embodiments of the invention and are not therefore to be considered to be limiting of its scope, the invention will be described and explained with additional specificity and detail through the use of the accompanying drawings in which:

[0011] Figure 1 is a block diagram that illustrates one example of a cross-site scripting attack;

[0012] Figure 2 illustrates a block diagram of a server computer that examines HTTP requests from user computers; and

[0013] Figure 3 illustrates examples of the data in an HTTP request that may contain script constructs.

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0014] The present invention relates to systems and methods for mitigating a cross-site scripting attack. Today, cross-site scripting attacks may permit attackers to access authentication tokens, access confidential information, execute client script in the security context of an origin web server, run commands with privilege on a user computer, and the like. Cross-site scripting attacks are prevalent and servers that are developed by inexperienced or careless developers are particularly susceptible to cross-site scripting attacks. Web servers that are susceptible to cross-site scripting attacks place the data of their users at risk.

[0015] The present invention helps reduce the risk of cross-site scripting attacks by examining incoming HTTP requests for potential script constructs that may indicate a cross-site scripting attack. In one embodiment, the present invention examines or evaluates portions of an HTTP request that may contain data derived from outside input. If a script construct is found in the HTTP request, then the HTTP request may be terminated or prevented from causing harm.

[0016] Figure 1 illustrates one example of a server computer 110 that is vulnerable to a cross-site scripting attack. This example of a cross-site scripting attack begins after an attacker computer 112 has sent an electronic message 102 to the user computer 100. The electronic message 102 includes an embedded link 104. The embedded link 104 typically includes a script. When the user clicks on the link 104, the script may be echoed back to the user computer 100 as HTML. In other words, the request sent to the server computer 110 causes the server computer 110 to send a script 108 back to the user computer 100. The user computer 100 executes the script 108 and sends, for example, sensitive information to the attacker computer 112.

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

[0017] A cross-site scripting attack typically occurs in a network environment where the server computer, the user computer, and the attacker computer are in communication over a network such as the Internet, a wide area network a local area network, and the like. In the example of Figure 1, the user computer 100, the server computer 110, and the attacker computer 112 are each connected with the Internet.

[0018] For instance, assume that a Stock Info Corporation maintains a web site that permits a user to track the user's stock prices. When the user logs into the web site, the server generates a dynamic web page that welcomes the user by name. The stock prices of the user are stored in a database and the credentials needed to access the database may be stored in a cookie on the user's computer.

[0019] In the context of Figure 1, an attacker may send an electronic message to the user that includes a link. The text of the link may be, for example, **www.mystocks.com/default.asp?name=<script>alert('Hello')</script>**. When the user selects this link, the user is trying to tell the server that the user's name is <script>alert('Hello')</script>. The web site would ordinarily display "Hello 'Name'" to the user. In this case, however, the web site may display "Hello " and run active content during rendering of the response.

[0020] In other words, the server generates a web page that contains the script of the attacker and serves that web page to the user computer. The user computer, however, recognizes and executes the script as if it came from a trusted source. The script may instruct the user computer to provide a cookie, which stores the user's stock information, to the attacker computer. The script, of course, can perform other activities as well.

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

[0021] Cross site scripting attacks can be prevented or mitigated by examining the HTTP request for active content. Active content includes, by way of example and not limitation, scripts, expressions, events, object tags, and the like. The HTTP request is examined by search for markers such as script constructs or other markers of active content.

[0022] For example, when a user is browsing the Internet, server computers generate web pages that are served to the client computer. Some of the content of the web page is prepared in response to requests from the client computer. Often, a web page will contain both text and Hyper Text Markup Language (HTML) markup. In order to distinguish between the text and the HTML markup, there are special character or character combinations that are interpreted in special ways. A "<script>" tag, for example, introduces a script for a particular scripting language. In this context, the script construct may be "<script*". By searching the user input or user request for this particular script construct, a cross site scripting attack can be prevented. In this embodiment, the present invention searches the user input or the user request for script constructs or other markers of active content.

[0023] The following discussion illustrates examples of various markers that may indicate a cross site scripting attack. One of skill in the art can appreciate that the present invention can be used to search a request for markers or script constructs in different scripting languages and that the script constructs for a particular scripting language may be different from the script constructs of another scripting language. The markers including script constructs described herein are by way of example and not limitation.

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

[0024] When a request is received, the request is search for markers of active content. A server computer, for example, may maintain a list of markers of active content. The markers of active content in the list can be updated as needed or augmented with additional markers of active content. Existing markers of active content can also be deleted or made inactive. In other words, the ability of a server to recognize a cross-site scripting attack can be enhanced over time by refining the list.

[0025] Figure 2 is an illustration of one embodiment of the present invention that reduces or mitigates the risk of cross-site scripting attacks. This example is illustrated using HTTP, but one of skill in the art can appreciate that the present invention can be implemented in other protocols. This example also illustrates a search for script constructs, which are a type of marker of active content. In Figure 2, the user computer 200 sends an HTTP request 204 to a server computer 210. A script module 212 of the server computer 210 examines the HTTP request 204 for script constructs. If the script module 212 discovers script constructs in the HTTP request 204, then the server computer 210 can refuse to execute the HTTP request. Alternatively, the server computer 210 can also inform the user computer that a script construct has been discovered in the HTTP request 204 and ask that the user resubmit the HTTP request.

[0026] Figure 3 illustrates an example of potential HTTP requests that may be sent by a user computer to a server computer. The HTTP request 300 may include, but is not limited to, query strings 302, fields of an HTTP form 304, headers, 306, and cookies 308. The server computer examines or evaluates all areas where user input is introduced. Form variables, query string variables, URLs with key value pairs, headers, and the like are examples of areas in an HTTP request were user input may be

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

introduced. From the perspective of the server computer 210, the user input is one example of data that is derived from an outside source.

[0027] These HTTP requests are evaluated by the server computer 210 to determine if a script construct or other marker of active content is present in the HTTP requests. If a script construct or other marker is found, then the HTTP request is typically aborted in a safe way. An error event may be triggered that can be handled by an application or logged for administrative review.

[0028] For example, a query string 300 may have the form NAME1=VAL1 & NAME2 = VAL2. The user input in this example is data at risk for the injection of a script. The data at risk in this example is "VAL1" and "VAL2." The server computer may only examine the data at risk for script constructs in one embodiment. In such an embodiment, for example, in a query string, the server computer may only need to examine the data provided by the user.

[0029] In one embodiment, this is performed by searching for markers that match a particular pattern. The present invention not only searches for typical script constructs such as angle brackets, but also for script constructs or markers of active content that are only harmful when rendered inside of particular HTML elements. For example, the server computer may examine an HTTP request to find a "<" symbol followed by an A to Z character ("<[A..Z] . . ."). Another script construct is a "<" symbol followed by a "!" symbol ("<!"). The server computer also examines the HTTP request for "&#". This example can inject script using ascii characters.

[0030] Another type of script construct is related to events such as "onclick". The server attempts to match a pattern that may represent a script. The server, for example, may search for "on[a..z]*=". This can detect a script when the script is injected as an

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

event to a type. Another type of script construct occurs when a script is used as the value of a tag attribute. For instance, a script can be used as an anchor in a href attribute. This is detected by examining the HTTP request for "*script:".

[0031] A script can also be injected using expressions. For instance, in an HTML page, a tag or button may include an expression that calculates the size of the element based on something else. This may be implemented using an expression by stating, for example, "width: expression". A script can be injected in the expression. In many cases, the expression is already implemented as a script. The danger to users exists when the script is coming to the server computer as user input. The user has no guarantee about what actions are performed by the script.

[0032] The present invention extends to both methods and systems for mitigating cross-site scripting attacks. The embodiments of the present invention may comprise a special purpose or general-purpose computer including various computer hardware, as discussed in greater detail below. The computer discussed below can be, for example, a server computer or a client computer.

[0033] Embodiments within the scope of the present invention also include computer-readable media for carrying or having computer-executable instructions or data structures stored thereon. Such computer-readable media can be any available media that can be accessed by a general purpose or special purpose computer. By way of example, and not limitation, such computer-readable media can comprise RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to carry or store desired program code means in the form of computer-executable instructions or data structures and which can be accessed by a general purpose or special purpose computer.

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

When information is transferred or provided over a network or another communications connection (either hardwired, wireless, or a combination of hardwired or wireless) to a computer, the computer properly views the connection as a computer-readable medium. Thus, any such connection is properly termed a computer-readable medium. Combinations of the above should also be included within the scope of computer-readable media. Computer-executable instructions comprise, for example, instructions and data which cause a general purpose computer, special purpose computer, or special purpose processing device to perform a certain function or group of functions.

[0034] The following discussion is intended to provide a brief, general description of a suitable computing environment in which the invention may be implemented. Although not required, the invention will be described in the general context of computer-executable instructions, such as program modules, being executed by computers in network environments. Generally, program modules include routines, programs, objects, components, data structures, etc. that perform particular tasks or implement particular abstract data types. Computer-executable instructions, associated data structures, and program modules represent examples of the program code means for executing steps of the methods disclosed herein. The particular sequence of such executable instructions or associated data structures represents examples of corresponding acts for implementing the functions described in such steps.

[0035] Those skilled in the art will appreciate that the invention may be practiced in network computing environments with many types of computer system configurations, including personal computers, hand-held devices, multi-processor systems, microprocessor-based or programmable consumer electronics, network PCs, minicomputers, mainframe computers, and the like. The invention may also be

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

practiced in distributed computing environments where tasks are performed by local and remote processing devices that are linked (either by hardwired links, wireless links, or by a combination of hardwired or wireless links) through a communications network. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

[0036] The present invention may be embodied in other specific forms without departing from its spirit or essential characteristics. The described embodiments are to be considered in all respects only as illustrative and not restrictive. The scope of the invention is, therefore, indicated by the appended claims rather than by the foregoing description. All changes which come within the meaning and range of equivalency of the claims are to be embraced within their scope.

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111